Identification Numbers and Check Digit Schemes

Trevor O'Brien

Fall 2018

MA 398, Dr. Molitierno

Faculty Mentor: Dr. Gopalakrishnan

Abstract

Identification numbers are all around us in our everyday life; from products in the supermarket to the airline tickets that we purchase to fly across the world. We are even given our own identification numbers: through things like our driver licenses and social security numbers! But what would happen if the number was in fact incorrect? The ramifications can be minimal, or potentially serious. How then, are we able to prevent this from happening? We prevent this from happening with the use of check digit schemes, which help prevent transmission errors of any kind. In this paper, we will learn about the formulation of identification numbers, and how check digit schemes are used to prevent any errors from occurring in the transmission process. We will also see how efficient the schemes are, and where exactly their flaws lie in terms of error prevention.

1 Introduction to Identification Numbers

Many times in our life have we encountered situations that require us to look up information. This can prove to be tedious, especially if the database being accessed is not proficient in organization. There may also be instances where we need to identify specific people or products to deal with problems that may have come up due to any number of scenarios. Once again, this can prove to be tedious and difficult without the proper method.

In order to deal with these challenges, sets of numbers known as "identification numbers" have been developed to help minimize the issues. They help in a variety of ways, being representative of specific products, social security numbers, driver's license numbers, and credit card numbers; just to name a few. Because these numbers are relatively short in length, they can be easily stored and retrieved by our computers to help us in our daily lives of information tracking. This helps to eliminate issues that may occur with cluttered databases or long retrieval processes.

Identification numbers are constructed through a process called a "hashing function". In a literal definitive sense, it is the process of taking information and converting it into an identification number. A simple example of this would be taking the first four characters of a person's last name and converting it into an identification number. So for the last name "Clark", the identification number

would be CLAR. This can also work for names with less than four letters: the last name Zia would be converted as ZIA*, with the star filling in for the last letter.

However, there are plenty of instances where the same identification number is assigned to multiple pieces of information. When this occurs, we call it "collision". Looking at our example above, if there are two people with the last name "Clark", they are going to be assigned the identification number CLAR. This can prove to be problematic if the identification number is used for a driver license or for a purchasing code for a product being delivered. Of course, to prevent issues like this from occurring, identification numbers are usually longer than four digits, and some numbers are broken up in subsets, which allow for a greater variety of identification numbers to be formed.

2 Transmission Errors

Given the vast array of identification numbers being formed, there are countless opportunities for errors to occur within the transmission process. This may happen for whatever reason, and certain transmission errors occur because of it. They are shown in the table below:

TABLE 1.2
Common Error Patterns

Error Type	Form	Relative Frequency
Single digit	a → b	79.1%
Transposition of adjacent digits	ab → ba	10.2%
Jump transposition	abc → cba	0.8%
Twin	aa → bb	0.5%
Phonetic	a0 ↔ 1a*	0.5%
Jump twin	aca → bcb	0.3%

*For a = 2, . . . , 9.

For the purposes of this paper, we will be focusing on the first two error types: single digit errors and transposition-of-adjacent-digit errors. This is because of their relative frequencies, as these two error types occur approximately ninety percent of the time. Single digit errors occur when one of the digits in the identification number changes to a different number in transmission, as shown from the form in the table. Transposition-of-adjacent-digit errors occur when two different side-by-side digits change places. Once again, this is shown form the form in the table. We will evaluate these two transmission errors through examples and with our check digit schemes, to see exactly how well they are at catching these particular errors.

3 Preliminaries

Before getting into the check digit schemes themselves, it is necessary to discuss some mathematical concepts relative to the rest of the paper. Some of the theorems presented here will be used in context of later proofs, while some concepts will be described to help the overall understanding of the way in which check digit schemes operate. First, it is beneficial to discuss prime numbers, as they are used extensively in the coming proofs. These numbers have to be greater than 1, and the only positive numbers that divide it are 1 and itself. Again, these numbers will prove substantial toward proving the errors within our schemes. Much of the mathematics behind our check digit schemes use the division algorithm, which is stated below:

For any two such integers x and y, where $Y > 0$, $\exists$ unique integers q and r such that $x = qy + r$, where $0 \leq r < (y$ - $1)$.

This definition plays an integral part in modular arithmetic, which all of schemes are based off of. This will be discussed shortly.

THEOREM 3.1 *For natural numbers, $a, b \geq 2, \exists$ integers s and t such that $gcd(a, b) = as + bt$.*

This is known as a linear combination of a and b. It will be used later in our proof of our check digit scheme for Universal Product Codes.

THEOREM 3.2 *If $a \mid b$, then $a \mid bc$.*

Proof: Let $b = ak$ for some k. Then $bc = akc$. □

Stemming from our previous theorem, Theorem 3.2 allows for us to divide any multiple of a number b by a, which is useful computing larger numbers in modular arithmetic.

THEOREM 3.3 *If $a \mid x$ and $a \mid y$, then $a \mid xy$.*

Proof: Let $a = xk$ for some k and $y = al$ for some l. Then $x+y = ak+al = a(k+l)$ where $k + l \in Z$. Thus $a \mid xy$. □

Again, we have another extension of our previous theorems, allowing for us to combine and divide more numbers in whatever "mod" we have.

THEOREM 3.4 *If $a, b, m \in Z$ such that $m \mid ab$ and $gcd(a, m) = 1$, then $m \mid b$.*

Proof: Since $m \mid ab$, it follows that $ab = mx$ for some integer x. By Theorem 3.1, since 1 is the *gcd* of integers a and m, we can write 1 as $1 = as + mt$ for some integers s, t. Multiplying the entire equation by b, we have $b = (ab)s + bmt = (mx)s + bmt = m(xs + bt)$. Since $(xs + bt) \in Z$, $m \mid b$. □

This theorem will be used to help prove our Universal Product Code scheme, similar to Theorem 3.1.

Before we continue to our types of identification numbers, we need to discuss modular arithmetic. The method for constructing modular equations is shown below:

Let x and n be integers with $n > 0$. Then $x \pmod{n} = r$.

This equation is closely related to the division algorithm mentioned previously, as the r in our equation above is the remainder that would be found in the algorithm. This will be the number that we will be paying particular attention to in our schemes, as the remainder will tell us whether we have encountered an error. But as we will see, the error does not always get caught.

4 US Postal Money Orders

The Unites States postal office uses an identification number system for their postal money orders. These numbers are designed as an 11-digit identification number, which allows for several different combinations to be created. The identification number is broken up into two separate parts: the first ten digits are known as the document number, while the last digit is the check digit.

Let $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}$ be the ten-digit document number associated with a US postal money order. The check digit, a_{11}, is determined by

$a_{11} = (a_1 + a_2 + \ldots + a_9 + a_{10}) (\text{mod } 9)$.

To put this into context, let us consider a US postal order with the identification number 58312044178. Based off of our definition, $a_1, a_2, \ldots, a_9, a_{10} = 5831204417$, and our check digit $a_{11} = 8$. After plugging our numbers into our equation, we can see that $(a_1, +a_2, \ldots, a_9, a_{10}) (\text{mod } 9) = (5 + 8 + \ldots + 1 + 7) (\text{mod } 9) = 35$ $(\text{mod } 9) = 8$. This matches our check digit, which confirms that our identification number was likely transmitted correctly. We can observe a second identification number 62390026336. Again, we have $a_1, \ldots, a_{10} = 6239002633$, and our check digit,

$a_{11} = 6$. When we try to find our check digit, we see that

$(6+2+3+\ldots+6+3) = 31$ (mod 9), which equals 4. This does not equal our check digit of 6, so we know that the error was caught in transmission. The question, then, is how precise this check digit scheme is.

THEOREM 4.1 *The USPS check digit scheme will catch all single-digit errors except where* $9 \mid |a_i - a_i\prime|$*, where* $a_i\prime$ *is the new number after transmission.*

Proof: Let $a_1, a_2, \ldots, a_9, a_{10}$ be the 10-digit document number associated with a US postal money order. Let the check digit a_{11} be determined by $a_{11} = (a_1 + a_2 + \ldots + a_i + \ldots + a_9 + a_{10}) (\text{mod } 9)$. Suppose the document number is transmitted as $a_1, a_2, \ldots, a_i\prime, \ldots, a_9, a_{10}$, where $a_i \neq a_i\prime$, and suppose the error is not caught. Then $a_{11} = (a_1 + a_2 + \ldots + a_i\prime + \ldots + a_9 + a_{10}) (\text{mod } 9)$. Looking at

both check digit equations, it follows that $(a_1+a_2+\ldots+a_i+\ldots+a_9+a_{10})(\text{mod } 9)$ $= (a_1+a_2+\ldots+a_i\prime+\ldots+a_9+a_{10})(\text{mod } 9)$. Since both equations are equivalent, it follows that $a_i(\text{mod } 9) = a_i\prime(\text{mod } 9)$. Thus $9 \mid |a_i - a_i\prime|$. □

Let us consider the identification number 27914009534. Through the definition that we have above, we can see that (2+7+...+5+3) (mod 9) = 40 (mod 9) = 4. Suppose the identification number is transmitted as 27014009534, where a_3 is transmitted incorrectly. We can see here that
(2+7+...+5+3) (mod 9) = 31 (mod 9) = 4. This shows that the error was not caught, even though the number was transmitted incorrectly. However, the only scenario in which the scheme is not caught is when a 9 is replaced by a 0, or vice versa, where it is not in the check digit position. The same cannot be said of the transposition-of-adjacent-digit errors, which do a significantly poorer job of catching transmission errors.

THEOREM 4.2 *The USPS check digit scheme will only catch transposition-of-adjacent-digit errors that involve the check digit.*

Proof: Let $a_1, a_2, \ldots, a_i, a_{i+1}, a_9, a_{10}$ be the 10-digit document number associated with a US postal money order. Let the check digit a_{11} be determined by $a_{11} = (a_1 + a_2 + \ldots + a_i + a_{i+1} + \ldots + a_9 + a_{10})(\text{mod } 9)$.

Case 1) Assume that the transposition does not involve the check digit. By

commutativity, we know that $a_{11} = (a_1 + a_2 + \ldots + a_{i+1} + a_i + \ldots + a_9 + a_{10})(\text{mod } 9)$. Thus the error will not be caught.

(Case 2) Assume that the transposition involves the check digit, and the error is not caught. WLOG, we know this to be a_{10} and a_{11} being transposed. Then $a_{10} = (a_1 + a_2 + \ldots + a_9 + a_{11})(\text{mod } 9)$. Note that $0 \leq a_{11} < 9$ and $0 \leq a_{10} \leq 9$. Subtracting the above equations gives us $a_{11} - a_{10} = (a_{10} - a_{11})(\text{mod } 9)$. By the definition of mod n, we have $9 \mid ((a_{11} - a_{10}) - (a_{10} - a_{11})) = 9 \mid (2a_{11}$ - $2a_{10})$ $= 9 \mid 2(a_{11} - a_{10})$. We know gcd (9,2) = 1. Then by Theorem 1.4 we know $9 \mid |a_{11} - a_{10}|$. Since $a_{11} \neq a_{10}$, it follows that $a_{11} - a_{10} \neq 0$. Thus the only solution for $9 \mid 2(a_{11} - a_{10})$ is when $a_{11} = 0$ and $a_{10} = 9$. This is invalid because 9 cannot be a check digit under (mod 9). This contradicts the assumption that the error was not caught. □

Because using an identification number that does not involve the check digit will not catch the error, let us consider the number 98765234510. Using our definition, we have (9+8+...+5+1) (mod 9) = 45 (mod 9) = 0. If we switch the first two numbers, we get a new identification number of 89765234510, which will also yield 45 (mod 9) = 0. As stated in the proof, since addition is commutative, switching the places of consecutive numbers will not alter the summation of the ten terms; thus, we can clearly see that there is a greater chance of not finding an

error that occurs in the transmission process.

5 Universal Product Codes

Perhaps the most well-known, or if not well-known, the most used identification would be the Universal Product Codes, or UPCs. Most of what is sold in stores utilize a UPC, which is the number located either above or below the bar code of the product. This identification number is only 12 numbers, and with the abundance of products in stores, there is a higher potential for collision to occur. Because of this, a UPC is divided into different sections to help prevent collision from occurring. The first number, a_1, refers to the "number system character". Essentially, this identifies the type of product being sold, such as general groceries or coupons. The second set of numbers, a_2, a_3, a_4, a_5, a_6, identifies the manufacturer that the product came from. The third set, $a_7, a_8, a_9, a_{10}, a_{11}$, identifies specifically what the product is. Lastly, the final number a_{12} is the check digit. While some companies have shortened their manufacturer set to four digits to deal with more products being sold, we will focus on the first set of numbers listed above. The definition, then, for the check digit scheme of UPCs is as follows:

Given the 11-digit number $a_1 - a_2, a_3, a_4, a_5, a_6 - a_7, a_8, a_9, a_{10}, a_{11}$, the check digit a_{12} satisfies the following equation: (3,1,3,1,3,1,3,1,3,1,3,1)

• $(a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}, a_{11}, a_{12}) = 0 \pmod{10}$.

As we can see, this check digit scheme uses the dot product to calculate any transmission errors. This takes two sequences of numbers that are equal length and produces a single number from it. Each "place" of one number sequence is multiplied by the same "place" in the other number sequence, and the summation of all the products is calculated. The result, in this context, will be used to determine our check digit.

To see this more clearly, let us take the UPC 0-53600-10054-0. If we expand this UPC into our equation, we see that
(3,1,3,1,3,1,3,1,3,1,3,1)•(0,5,3,6,0,0,1,0,0,5,4)= 0 (mod 10). If we expand the left side of the equation, we get (0+5+9+6+0+0+3+0+0+5+12+0) = 0 (mod 10). This simplifies to $40 \equiv 0 \pmod{10}$, so our check digit scheme shows the number was transmitted correctly. The use of "mod 10" allows for more precise error calculations than our previous two check digit schemes for single-digit errors, because each remainder in "mod 10" can only occur if it is the digit of the ones place. To explain in another way, $49 \pmod 7 \equiv 21 \pmod 7$, where 9 and 1 are both in the ones place; this cannot happen under "mod 10".

THEOREM 5.1 *The Universal Product Code (UPC) check digit scheme will catch all single-digit errors.*

Proof: Let $a_1, a_2, \ldots, a_i, \ldots, a_{11}, a_{12}$ be the 12-digit UPC where $1 \leq i \leq 12$. Thus we have the equation $(3, 1, 3, 1, \ldots, 3, 1)(a_1, a_2, \ldots, a_i, \ldots, a_{11}, a_{12}) = \pmod{10}$.

Suppose the UPC is transmitted as $a_1, a_2, \ldots, a_i\prime, \ldots, a_{11}, a_{12}$, where $a_i \neq a_i\prime$, and suppose the error is not caught. Then $(3,1,3,1,\ldots,3,1)(a_1, a_2, \ldots, a_i\prime, \ldots, a_{11}, a_{12})$ $= (\text{mod } 10)$. Looking at both check digit equations, we have

$(3,1,3,1,\ldots,3,1)(a_1, a_2, \ldots, a_i, \ldots, a_{11}, a_{12}) = (\text{mod } 10)$ and

$(3,1,3,1,\ldots,3,1)(a_1, a_2, \ldots, a_i\prime, \ldots, a_{11}, a_{12}) = (\text{mod } 10)$, which can be written as

$(3,1,3,1,\ldots,3,1)(a_1, a_2, \ldots, a_i, \ldots, a_{11}, a_{12})$ - $(3,1,3,1,\ldots,3,1)(a_1, a_2, \ldots, a_i\prime, \ldots, a_{11}, a_{12})$ $= (\text{mod } 10)$.

(Case 1) Assume that a_i and $a_i\prime$ are multiplied by 3. Thus

$(3,1,\ldots,3,1)(a_1, \ldots, a_i, \ldots, a_{12})$ - $(3,1,\ldots,3,1)(a_1, \ldots, a_i\prime, \ldots, a_{12})(\text{mod } 10) = 0$

which yields

$(3a_1+1a_2+\ldots+3a_i+\ldots+3a_{11}+1a_{12})-(3a_1+1a_2+\ldots+3a_i\prime+\ldots+3a_{11}+1a_{12})(\text{mod } 10) = 0$ which yields

$3a_1+1a_2+\ldots+3a_i+\ldots+3a_{11}+1a_{12}-3a_1-1a_2-\ldots-3a_i\prime-\ldots-3a_{11}-1a_{12}(\text{mod } 10) = 0$. After cancellation we have $3a_i-3a_i\prime(\text{mod } 10) = 0$, which can be rewritten as $3(a_i - a_i\prime)(\text{mod } 10) = 0$. This implies that $10 \mid 3(a_i - a_i\prime)$. By Theorem 3.4, $10 \mid (a_i - a_i\prime)$. Since $a_i \neq a_i\prime$, it follows that $a_i - a_i\prime \neq 0$. We also know $0 < |a_i - a_i\prime| \leq 9$ since $0 \leq a_i, a_i\prime \leq 9$. This contradicts the assumption that the error was not caught.

(Case 2) Assume both a_i and $a_i\prime$ are multiplied by 1. Thus

$(3,1,\ldots,3,1)(a_1, \ldots, a_i, \ldots, a_{12})$ - $(3,1,\ldots,3,1)(a_1, \ldots, a_i\prime, \ldots, a_{12})(\text{mod } 10) = 0$

which yields

$(3a_1+1a_2+\ldots+1a_i+\ldots+3a_{11}+1a_{12})-(3a_1+1a_2+\ldots+1a_i\prime+\ldots+3a_{11}+1a_{12})(\text{mod } 10) = 0$ which yields

$3a_1+1a_2+\ldots+1a_i+\ldots+3a_{11}+1a_{12}-3a_1-1a_2-\ldots-1a_i\prime-\ldots-3a_{11}-1a_{12}(\text{mod } 10) = 0$. After cancellation we have $(1a_i - 1a_i\prime)(\text{mod } 10) = 0$, which can be rewritten as $1(a_i - a_i\prime)(\text{mod } 10) = 0$. Thus $(a_i - a_i\prime) = 0 \text{ (mod } 10)$. Since $a_i \neq a_i\prime$, it follows that $a_i - a_i\prime \neq 0$. We also know that $0 \leq a_i, a_i\prime \leq 9$. Thus $a_i - a_i\prime$ can never be a multiple of 10. This contradicts the assumption that the error was not caught. □

As stated above, if we take any UPC scheme possible, and there is a single-digit error associated with it, then it will be caught every time. Considering that the goal of the schemes are to catch as many errors as possible, this is certainly the best we can have.

THEOREM 5.2 *The UPC check digit scheme will catch all transposition-of-adjacent-digit errors except where* $|a_i - a_i\prime| = 5$.

Proof: Let $a_1, a_2, \ldots, a_i, a_{i+1}, \ldots, a_{11}, a_{12}$ be the 12-digit UPC where $1 \leq i \leq 12$. Thus we have the equation $(3, 1, \ldots, 3, 1)(a_1, \ldots, a_i, a_{i+1}, \ldots, a_{12}) = 0 \text{ (mod } 10)$. Suppose the UPC is transmitted as $a_1, a_2, \ldots, a_{i+1}, a_i, \ldots, a_{11}, a_{12}$ and suppose the error is not caught. Thus $(3, 1, \ldots, 3, 1)(a_1, \ldots, a_{i+1}, a_i, \ldots, a_{12}) = 0 \text{ (mod } 10)$. These equations can be rewritten as $(3, 1, \ldots, 3, 1)(a_1, \ldots, a_i, a_{i+1}, \ldots, a_{12}) -$

$(3, 1, \ldots, 3, 1)(a_1, \ldots, a_{i+1}, a_i, \ldots, a_{12}) = 0 \pmod{10}$.

(Case 1) Suppose i is odd. Expanding the equation, we have

$3a_1 + 1a_2 + \ldots + 3a_i + 1a_{i+1} + \ldots + 3a_{11} + 1a_{12} - 3a_1 - 1a_2 - \ldots - 3a_{i+1} - 1a_i - \ldots - 3a_{11} - 1a_{12} \pmod{10} = 0$. After cancellation we have

$3a_i + 1a_{i+1} - 3a_{i+1} - 1a_i \pmod{10} = 0$ which simplifies to

$2a_i - 2a_{i+1} \pmod{10} = 0$ which simplifies to

$2(a_i - a_{i+1}) \pmod{10} = 0$. Thus $2(a_i - a_{i+1}) = 0 \pmod{10}$. This can be written as $10 \mid 2|a_i - a_{i+1}|$, which is equivalent to $5 \mid |a_i - a_{i+1}|$. Since $a_i \neq a_{i+1}$, it follows that $a_i - a_{i+1} \neq 0$. We also know that $0 \leq a_i, a_{i+1} \leq 9$, so $0 < |a_i - a_{i+1}| \leq 9$. Because 5 is prime, the only number between 0 and 9 that is divisible by 5 is 5 itself. Therefore, $|a_i - a_i\prime| = 5$.

(Case 2) The case for when i is even follows a similar proof. □

While this scheme is not as successful as single-digit errors, it still does a fairly good job. Say we have a UPC 5-02003-91562-1, and it is transmitted as 0-52003-91562-1. Here we see that a_1 and a_2 are switched during transmission. Using our formula, we find that (3,1,. . . ,3,1)•$(0, 5, \ldots, 2, 1) = 0 \pmod{10} \equiv 70 = 0 \pmod{10}$. Although in this instance the error was not caught, the scheme is still able to catch most errors, while being able to catch all single-digit errors.

6 International Standard Book Numbers

Every book that is printed has a 10-digit identification number. This is known as an International Standard Book Number, or ISBN. The first number in the ISBN is known as the "group" or "country" number, which identifies the language area and the nation or geographic grouping of the publisher. The second set of numbers identifies the publisher. This number can be any length of numbers, but will typically be between two and five numbers. The third set of numbers is the book code that has been chosen by the publisher, which again can vary in length. This set of numbers will usually depend on the publisher code, as a longer publisher code will result in less book code options. The last number is the check digit. We will look at the ISBN number 3-357-02001. This will be calculated using the formula found in the following definition:

Given the 10-digit ISBN $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}$, the check digit, a_{10}, is determined by the equation

(10,9,8,7,6,5,4,3,2,1)•($a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}$) = 0 (mod 11). If the check digit a_{10} happens to be 10, the letter X is used instead.

Similar to our previous scheme, we are using the dot product to calculate if an error has occurred. Using the ISBN mentioned above, we can calculate the following: (10,9,8,7,6,5,4,3,2,1)•$(3, 3, 5, 7, 0, 2, 0, 0, 1, 4) = 0$(mod 11). If we expand the left side of the equation, we see that 30+27+40+49+0+10+0+0+2+4 = 0

(mod 11) which simplifies to $162 = 0 \pmod{11}$. This statement is false, since $162 = 8 \pmod{11}$; thus showing us that the ISBN was transmitted incorrectly. Even with a check digit of 10, the ISBN can still be valid under "mod 11", which is different than previous schemes that we have seen.

THEOREM 6.1 *The International Standard Book Number (ISBN) check digit scheme will catch all single-digit and transposition-of-adjacent-digit errors.*

Proof: First we will consider a single-digit error. Let $a_1, \ldots, a_i, \ldots, a_{10}$ be the 10-digit ISBN with $1 \leq i \leq n$. Thus we have $(10, 9, \ldots, 2, 1)(a_1, \ldots, a_i, \ldots, a_{10}) = 0 \pmod{11}$. Suppose the ISBN is transmitted as $a_1, \ldots, a_i\prime, \ldots, a_{10}$, where $a_i \neq a_i\prime$, and suppose the error is not caught. Then $(10, 9, \ldots, 2, 1)(a_1, \ldots, a_i\prime, \ldots, a_{10}) = 0 \pmod{11}$. Looking at both equations, we can rewrite them as $(10, \ldots, 1)(a_1, \ldots, a_i, \ldots, a_{10}) - (10, \ldots, 1)(a_1, \ldots, a_i\prime, \ldots, a_{10}) = 0 \pmod{11}$. Since we are computing a single-digit error, we know that a_i and $a_i\prime$ will be multiplied by the same integer k, where $1 \leq k \leq 10$. By substituting k into the equation, we have $(10, \ldots, k, \ldots, 1)(a_1, \ldots, a_i, \ldots, a_{10}) - (10, \ldots, k, \ldots, 1)(a_1, \ldots, a_i\prime, \ldots, a_{10}) = 0 \pmod{11}$

$(10a_1 + \ldots + ka_i + \ldots + 1a_{10}) - (10a_1 + \ldots + ka_i\prime + \ldots + 1a_{10}) \pmod{11} = 0$

$10a_1 + \ldots + ka_i + \ldots + 1a_{10} - 10a_1 - \ldots - ka_i\prime - \ldots - 1a_{10} \pmod{11} = 0.$

After cancellation we have $(ka_i - ka_i\prime) \pmod{11} = 0$, which is equivalent to $k(a_i - a_i\prime) \pmod{11} = 0$. Thus $k(a_i - a_i\prime) = 0 \pmod{11}$. This implies that

$11 \mid k(a_i - a_i\prime)$. Since $1 \leq k \leq 10$ and 11 is prime, the gcd (11,k) = 1. By Theorem 1.4, we know

$11 \mid (a_i - a_i\prime)$. Since $a_i \neq a_i\prime$, it follows that $a_i - a_i\prime \neq 0$. We also know $0 \leq a_i, a_i\prime \leq 9$.

Hence $0 < |a_i - a_i\prime| \leq 9$. This implies that $11 \not| |a_i - a_i\prime|$. This contradicts the assumption that the error was not caught.

Next, let us consider a transposition-of-adjacent-digit error.

Let $a_1, \ldots, a_i, a_{i+1}, \ldots, a_{10}$ be the 10-digit ISBN with $1 \leq i \leq n$. Thus we have $(10, 9, \ldots, 2, 1)(a_1, \ldots, a_i, a_{i+1}, \ldots, a_{10}) = 0 \pmod{11}$. Suppose the ISBN is transmitted as $a_1, \ldots, a_{i+1}, a_i, \ldots, a_{10}$, where $a_i \neq a_i\prime$, and suppose the error is not caught. Then $(10, 9, \ldots, 2, 1)(a_1, \ldots, a_{i+1}, a_i, \ldots, a_{10}) = 0 \pmod{11}$. Looking at both equations, we can rewrite them as

$(10, 9, \ldots, 2, 1)(a_1, \ldots, a_i, a_{i+1}, \ldots, a_{10}) - (10, 9, \ldots, 2, 1)(a_1, \ldots, a_{i+1}, a_i, \ldots, a_{10}) = 0 \pmod{11}$. Since we are computing a transposition-of-adjacent-digit error, we know that a_i and a_{i+1} will be multiplied by the same adjacent integer k and m, where $1 \leq m < k \leq 10$. By substituting k and m into the equation, we have

$(10, \ldots, k, m, \ldots, 1)(a_1, \ldots, a_i, a_{i+1}, \ldots, a_{10}) - (10, \ldots, k, m, \ldots, 1)(a_1, \ldots, a_{i+1}, a_i, \ldots, a_{10}) \pmod{11} = 0$ which yields

$(10a_1 + \ldots + ka_i + ma_{i+1} + \ldots + 1a_{10}) - (10a_1 + \ldots + ka_{i+1} + ma_i + \ldots + 1a_{10}) \pmod{11} = 0$ which yields

$10a_1 + \ldots + ka_i + ma_{i+1} + \ldots + 1a_{10} - 10a_1 - \ldots - ka_{i+1} - ma_i - \ldots - 1a_{10} (\text{mod } 11) = 0$. After cancellation we have $(ka_i + ma_{i+1} - ka_{i+1} - ma_i) (\text{mod } 11) = 0$, which can be reduced to $(k-m)(a_i - a_{i+1}) (\text{mod } 11) = 0$. Thus $(k-m)(a_i - a_{i+1}) \equiv 0 \text{ (mod } 11)$. This implies that $11 \mid (k-m)(a_i - a_{i+1})$. Since $1 \leq m < k \leq 10$, then $0 < k - m < 10$. Since 11 is prime, the gcd $(11, k-m) = 1$. Therefore, by Theorem 1.4, we know $11 \mid (a_i - a_{i+1})$. We also know $a_i - a_{i+1} \neq 0$ since $a_i \neq a_i\prime$ and $0 \leq a_i, a_{i+1} \leq 9$. This implies that $11 \not| (a_i - a_{i+1})$. This contradicts the assumption that the error was not caught. □

We can finally see a check digit scheme that is able to catch every single-digit and transposition-of-adjacent-digit error. However, there are two problems that can arise while using this system. The first issue arises with the use of the letter X. While the scheme is successful even while using the letter X, it is much better to have the check digit between 0 and 9. This prevents us from having to introduce new characters into our schemes; thus allowing for us to keep identification numbers that are made up entirely of numbers. The second problem is that while the scheme is extremely successful in catching transmission errors, it only works for identification numbers of length 10. While these concerns may be a bit narrow-minded, it keeps the focus up to find schemes that are able to satisfy all potential issues that may arise in the transmission process.

7 Conclusion

While there remains to be a host of potential fallout regarding check digit schemes, the fact remains that this particular method for identifying products is fairly sound. While there are some cracks within catching every error, using modular arithmetic for the check digit schemes provides us with a simple and efficient strategy. As discussed at the beginning of the paper, it is important to have efficiency when gathering data on large scales, such as the ones presented here. Perhaps other areas of mathematics could prove better in terms of error prevention rate and overall efficiency numbers; but the math laid out in this paper provides sufficient evidence to warrant continuation with check digit schemes. Maybe it will cost us an extra 10 dollars at the grocery store, or getting placed on the wrong flight; but the overall success of the check digit schemes is understood, and will continue to be largely utilized for the foreseeable future.

References

[1] J. Kirtland. Identification Numbers and Check Digit Schemes. *The Mathematical Association of America*, 2001.